JASPER REPORT - REPORT DESIGN

http://www.tutorialspoint.com/jasper_reports/jasper_report_designs.htm

The JRXML templates (or JRXML files) in JasperReport are standard XML files, having an extension of .jrxml. All the JRXML files contain tag <jasperReport>, as root element. This in turn contains many sub-elements (all of these are optional). JasperReport framework can handle different kinds of data sources. In this tutorial we shall show how to generate a basic report, just by passing a collection of Java data object (using Java beans), to the JasperReport Engine. The final report shall display Name and Country of a list of people.

This chapter covers on how to design a JasperReport. Following steps are covered in this chapter:

- Creating a JRXML Report Template.
- Previewing the XML Report Template.

Creating a JRXML Report Template

Create the JRXML file which is **jasper_report_template.jrxml** using a text editor and save this file in *C:\tools\jasperreports-5.0.1\test* as per our environment setup.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jasperReport PUBLIC "//JasperReports//DTD Report Design//EN"
"http://jasperreports.sourceforge.net/dtds/jasperreport.dtd">
<jasperReport xmlns="http://jasperreports.sourceforge.net/jasperreports"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://jasperreports.sourceforge.net/jasperreports
http://jasperreports.sourceforge.net/xsd/jasperreport.xsd"
name="jasper_report_template" language="groovy" pageWidth="595"
pageHeight="842" columnWidth="555" leftMargin="20" rightMargin="20"
topMargin="20" bottomMargin="20">
    <queryString>
 <![CDATA[]]>
    </queryString>
    <field name="country" >
        <fieldDescription><![CDATA[country]]></fieldDescription>
    </field>
    <field name="name" >
        <fieldDescription><![CDATA[name]]></fieldDescription>
    </field>
    <columnHeader>
     <band height="23">
 <staticText>
     <reportElement mode="Opaque" x="0" y="3" width="535"
   height="15" backcolor="#70A9A9" />
     <box>
         <bottomPen lineWidth="1.0" lineColor="#CCCCCC" />
     </box>
     <textElement />
     <text><![CDATA[]]> </text>
 </staticText>
 <staticText>
     <reportElement x="414" y="3" width="121" height="15" />
     <textElement textAlignment="Center"
         verticalAlignment="Middle">
  <font isBold="true" />
     </textElement>
     <text><![CDATA[Country]]></text>
 </staticText>
 <staticText>
     <reportElement x="0" y="3" width="136" height="15" />
     <textElement textAlignment="Center"
         verticalAlignment="Middle">
  <font isBold="true" />
     </textElement>
     <text><![CDATA[Name]]></text>
 </staticText>
     </band>
    </columnHeader>
    <detail>
        <band height="16">
  <staticText>
      <reportElement mode="Opaque" x="0" y="0" width="535"
   height="14" backcolor="#E5ECF9" />
      <box>
             <bottomPen lineWidth="0.25" lineColor="#CCCCCC" />
      </box>
      <textElement />
      <text><![CDATA[]]> </text>
  </staticText>
  <textField>
      <reportElement x="414" y="0" width="121" height="15" />
      <textElement textAlignment="Center"
          verticalAlignment="Middle">
   <font size="9" />
      </textElement>
      <textFieldExpression >
          <![CDATA[$F{country}]]>
  </textFieldExpression>
  </textField>
  <textField>
      <reportElement x="0" y="0" width="136" height="15" />
      <textElement textAlignment="Center"
          verticalAlignment="Middle" />
          <textFieldExpression >
   <![CDATA[$F{name}]]>
          </textFieldExpression>
  </textField>
        </band>
    </detail>
</jasperReport>
```

Here are the details of main fields in the above report template:

- <queryString>: This is empty (as we are passing data through Java Beans). Usually contains the SQL statement which retrieves the report result.
- <field name>: This element is used to map data from datasources or queries into report templates.*name* is re-used into the report body (they are case-sensitive).
- <fieldDescription>: This element maps the field name with the appropriate element in the XML file.
- <staticText>: This defines static text that does not depend on any datasources, variables, parameters, or report expressions.
- <textFieldExpression>: This defines the appearance of result field.
- $F{country}: This is a variable that contains the value of result predefined field in the tag <field name>.
- <band>: Bands contain the data that is displayed in the report.

Once the report design is ready, save it in C:\ directory.

Previewing the XML Report Template

There's a utility *net.sf.jasperreports.view.JasperDesignViewer* available in JasperReports JAR file which helps preview a report design without having to compile or fill it. This utility is a standalone Java application, hence can be executed using ANT.

Let's write an ANT target **viewDesignXML** to view the JRXML. So let's create and save **build.xml** under

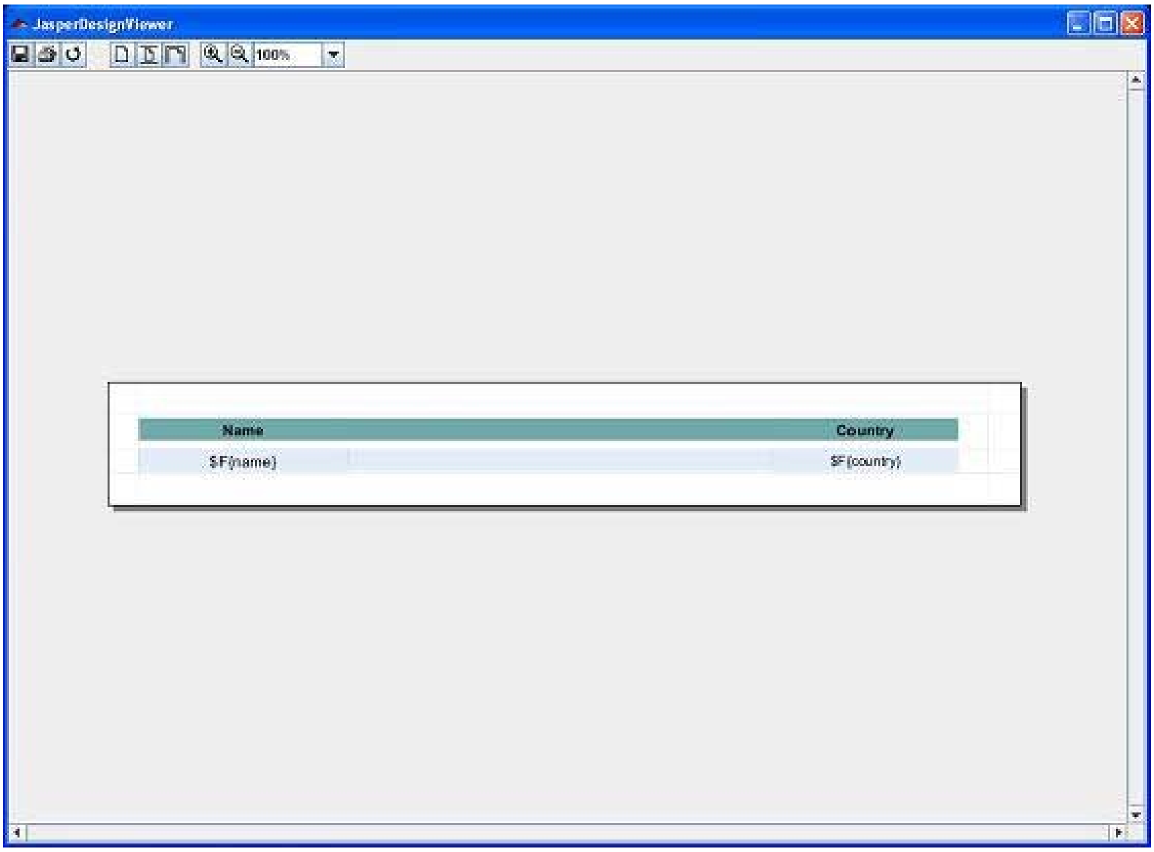
C:\tools\jasperreports-5.0.1\test directory (should be placed in the same directory where JRXML is placed). Here is the build.xml file:

```
<?xml version="1.0" encoding="UTF-8"?>
<project name="JasperReportTest" default="viewDesignXML" basedir=".">

    <import file="baseBuild.xml" />
    <target name="viewDesignXML" description="Design viewer is launched
 to preview the JXML report design.">
     <java classname= "net.sf.jasperreports.view.JasperDesignViewer"
     fork="true">
     <arg value="-XML" />
     <arg value="-F${file.name}.jrxml" />
     <classpath ref />
     </java>
    </target>

</project>
```

Next, let's open command prompt and go to the directory where build.xml is placed. Execute the command **ant** (As the viewDesignXML is the default target). Output is follows:

```
C:\tools\jasperreports-5.0.1\test>ant
Buildfile: C:\tools\jasperreports-5.0.1\test\build.xml

viewDesignXML:
[java] log4j:WARN No appenders could be found for logger
(net.sf.jasperreports.engine.xml.JRXmlDigesterFactory).
[java] log4j:WARN Please initialize the log4j system properly.
```

Log4j warning can be ignored, and as a result of above execution, a window labeled "JasperDesignViewer" opens displaying our report template preview.



As we see only report expressions for obtaining the data are displayed, as JasperDesignViewer doesn't have access to the actual datasource or report parameters. Terminate the JasperDesignViewer by closing the window or by hitting Ctrl-c in the command-line window.